

Hybrid Contribution of JPEG200 Video Files for Professional Production Centers Reliability and Non-Reliability Transmission File Based in Image Quality Needed

Jesus M. Barbero^{*}

OEI/EUI Technical University of Madrid, Ctra Valencia KM 7, 28031 Madrid, Spain

Received: 29 August 2011; Revised: 03 January, 2012; Accepted: 16 January, 2012

Abstract: ATM, SDH or satellite have been used in the last century as the contribution network of Broadcasters. However the attractive price of IP networks is changing the infrastructure of these networks in the last decade. Nowadays, IP networks are widely used, but their characteristics do not offer the level of performance required to carry high quality video under certain circumstances. Data transmission is always subject to errors on line. In the case of streaming, correction is attempted at destination, while on transfer of files, retransmissions of information are conducted and a reliable copy of the file is obtained. In the latter case, reception time is penalized because of the low priority this type of traffic on the networks usually has. While in streaming, image quality is adapted to line speed, and line errors result in a decrease of quality at destination, in the file copy the difference between coding speed vs line speed and errors in transmission are reflected in an increase of transmission time. The way news or audiovisual programs are transferred from a remote office to the production centre depends on the time window and the type of line available; in many cases, it must be done in real time (streaming), with the resulting image degradation. The main purpose of this work is the workflow optimization and the image quality maximization, for that reason a transmission model for multimedia files adapted to JPEG2000, is described based on the combination of advantages of file transmission and those of streaming transmission, putting aside the disadvantages that these models have. The method is based on two patents and consists of the safe transfer of the headers and data considered to be vital for reproduction. Aside, the rest of the data is sent by streaming, being able to carry out recuperation operations and error concealment. Using this model, image quality is maximized according to the time window.

In this paper, we will first give a briefest overview of the broadcasters' requirements and the solutions with IP networks. We will then focus on a different solution for video file transfer. We will take the example of a broadcast center with mobile units (unidirectional video link) and regional headends (bidirectional link), and we will also present a video file transfer file method that satisfies the broadcaster' requirements.

Keywords: Broadcast contribution networks, CDNs, JPEG2000, video contribution, video file transfer.

INTRODUCTION

The most important functions of a broadcast contribution network are as follows:

- For stored programs: the process of data transfer has to be achieved before a specific time (before the "in-time" play out of the video).
- For live events: the central station has to receive the image less than a maximum end-to-end delay because of the conversation mode.

The network has to achieve these two restrictions with the best image quality.

Classical Workflow

In classical workflow (with video links), the delivery of stored programs can be done by two ways:

- Physical transport of video tapes copies with all operational aspects associated with the dubbing process.
- Live video delivery with video links, coordination process is needed between origin and destination centers to properly specify the generic for 'play', 'record', and 'signal level test' operations.

When the tape is sent, image suffers no degradation (in digital video). However, when the image is sent by video link, the image quality must be adapted to the available link bandwidth, a clear example is when the program is edited in JPEG2000 [1] 50 Mbps and the Drive Away Digital Satellite News Gathering (DSNG) unit has a MPEG2 [2] 12 Mbps uplink. In addition to this, video streaming is strongly affected by errors.

Broadcast Contribution Network

It is possible to study the Broadcast Contribution Networks (BCNs) as can be seen in Table 1.

For file distribution over unidirectional links, there are various file transfer protocols based on retransmission patterns of the same file.

Address correspondence to this author at the OEI/EUI Technical University of Madrid, Ctra Valencia KM 7, 28031 Madrid, Spain; Tel: +3491 3367884; Fax: +3491 3367520; E-mail: Jesus.martinezb@upm.es

Table 1. Contribution Networks Types and Main Protocols.

Link Type	Operation	Protocols
Unidirectional	File delivery	ALC, NORM, FLUTE
	Live Transfer	DVB-H, DVB-T, etc..
Bidirectional	File Delivery	FTP, Aspera, etc
	Live Transfer	DVB o IP, MPEG2, etc.

Live video transfer over unidirectional links can be done with the well known Digital Video Broadcasting (DVB), with their different extensions for terrestrial (DVB-T) [3], satellite (DVB-S) [4], mobile (DVB-H) [5] or cable DVB-C [6].

Bidirectional links can support the File Transfer Protocol (FTP) [7] that is widely used for file transfers. The bidirectional Ethernet and IP are the most commonly used protocols worldwide. Gigabit Ethernet is now commonly implemented in local area networks and gains in popularity within the wide area networks.

A broadcaster may benefit from the latest advance in the IP world to interconnect its digital broadcast network with its local area network, and makes use of the IP backbone as a contribution network. Doing this, the broadcaster gains improved implementation costs, accessibility and programming customization.

For stored programs, physical transport of video tapes copies is replaced with file transfer. Live events can be transmitted in a real time with a QoS network.

The biggest problem with stored programs is when broadcaster needs the video file in a short period of time before the “on air time” (e.g. news). In addition to this, the production of professional video content generates huge data file which must be transmitted among different production centers.

Nowadays, Digital Satellite News Gathering (DSNG) units usually have a unidirectional link, so that the content has to be sent as live events by satellite link. Video streaming delivery requires the adjustment of image quality to the transmission speed. Compressed video file may be reproduced at an uncompressed bitrate in real time, the signal is codified into a bitstream of compressed digital video (typically MPEG2, 12 up to 20 Mbps) for its transmission through network as live event. Line transmission errors are concealed on reception using methods based on image correlation, depending on line noise; it is occasionally impossible to restore the image if important parts of it have been damaged. At destination, it is again transcoded to uncompressed video for codified to the compression format of the production center. The image loses quality when it is coded and transcoded, bit rates changes, errors in transmission and drops in the satellite links. These processes involve a great quantity of human as well as technical resources for video transfers.

The processes involved in a news transfer from remote sources to the central station include the following (as it can be seen in Fig. (1)):

- Capture of images and storing in compressed video formats.
- Image edition in computers.
- Decompression of images in uncompressed video (real time reproduction).
- Compression of uncompressed bitstream.
- Video Transmission over DVB-S.
- Decompression to Serial Digital Interface (SDI) video.
- Video compression.

These processes involve transcodifications, speed changes in compressions and error concealment due to transmission errors. All these techniques assume a large degradation of image quality; this degradation increases when production and transfer formats are different.

A solution to change the DSNG link to bidirectional with the assumption of extra link cost; a simple ftp can do the job. This solution has two serious drawbacks; ftp needs an exact copy of file (it's difficult to determine the transfer time due the line errors), and the link delay (ftp present a low bandwidth occupation with high delays).

Due to the transfer time problem prediction, broadcaster usually sends the urgent material as live events, in order to ensure that the program has been received before the playout in-time.

The intention of this paper is to propose and evaluate a flexible and simple multimedia transfer file method that preserves vital part of the file. We propose an approach to perform the file transfer with a predictable transfer time maximizing the image quality. The method is based on two patents [8, 9] and consists of the safe transfer of the headers and data considered to be vital for reproduction. Taking full advantages of image format and structure, this file transfer mechanism send part of file *via* ftp and the other part *via* streaming. With this method, Broadcasters avoid transcodification and base band conversion in video file transmissions.

The paper is structured as follows: Section 2 introduces the techniques developed for transfer file and an overview of JPEG2000; Section 3 presents the method; Section 4 presents the scenario; Section 5 presents several experimental results; and last section summarizes this work pointing out some conclusions.

BACKGROUND

In this section, we will review some aspect of professional video delivery.

Video Transmission Errors

Video streaming is strongly affected by errors and loses introduced by transmission systems, the problem of error control has become a crucial one due to the increasing use of wireless channels and packet-switched networks in distributed media services. Such channels are characterized by high error rate, error burst, packet loses, and temporal variability. Video error control procedures are targeted at

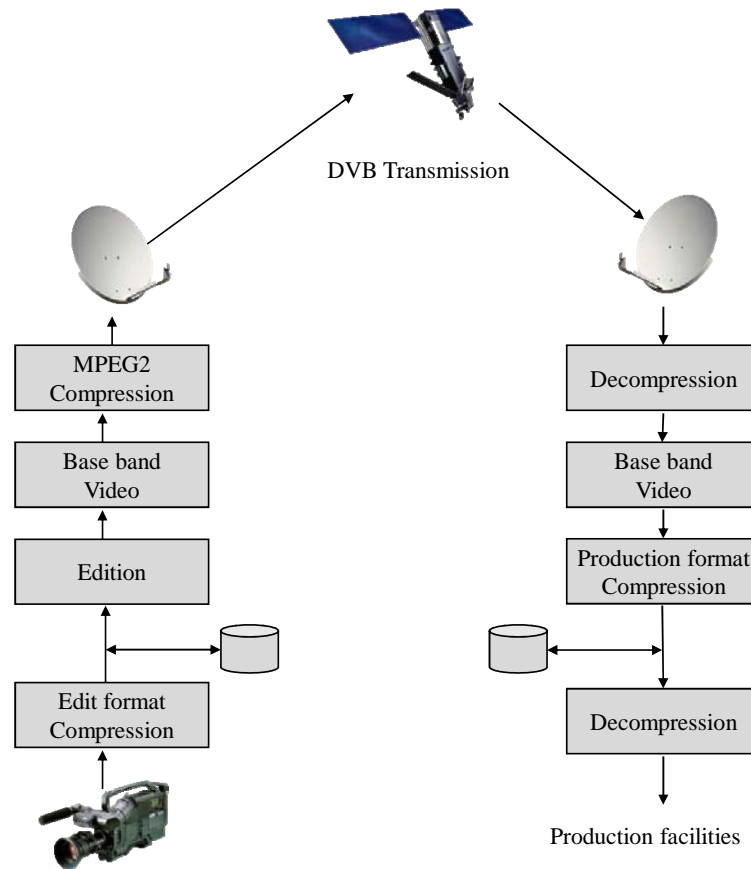


Fig. (1). Traditional video contribution workflow.

removing or reducing the impact of errors on the video quality. A good review on the subject can be found in [10].

These techniques can be divided into two families:

- Channel based techniques. The goal protects the message before transmitting it. These include Forward Error Correction codes (FECs) and retransmission protocols (ARQ).
- Source-based techniques, which exploit the data source characteristics to improve the reliability of the transmission. These include the error-resilient video coding, where the problem is handled before the transmission (a priori) and error concealment, in which the video is post processed in the decoder when the errors occurs (a posteriori).

In the last decades, there have been multiple efforts to combine these approaches with the aim of achieving the best video quality, but inspite of all these efforts, errors can still occur if the digital signal becomes excessively corrupted. Finally, the error can happen in a vital part of the data (e.g. headers), resulting the complete loss of the payload, that may cause the loss of large portions of stream. In order to solve this problem, we present an approach to preserve the biggest part of vital data during the transmission and the storage of video files.

File Transfer

Tapeless contribution workflow means that content producers and broadcasters move large files over long

distances (rushes to postproduction facilities, finished programs to playout facility, exchange between broadcasters, news bureaux exchange, sports clip, etc.).

Bidirectional Links

High-performance WANs with HTTP, FTP [11], etc. use Transmission Control Protocol (TCP) to guarantee the transfer using acknowledgements and if necessary retransmissions and a “sliding window” flow control mechanism. Originally TCP only supported a maximum window size of 64 kB, which limits the throughput, for example 100 Mbit/s for a Round-Trip Time of 5 ms [12]. On a congested network, the effects of packet loss mean that the traditional ftp algorithm performs poorly.

In order to improve the performance for media file transfer, a number of approaches exist:

- Commercial products (e.g. Signiant, Aspera, FileCatalyst, etc.) are often based on User Datagram Protocol (UDP) instead of TCP, and use their own proprietary flow control and congestion control algorithms that are optimized for large file transfer.
- Open-source alternatives offer some good performance and several features available from commercial software, but with a generally ‘bare-bones’ interface. Some examples are: UDP-based protocols (UDT, UFTP [13], Tsunami, VFER) ; TCP-based protocols (FDT, bbFTP, bbCP, GridFTP).

In [14], the authors experimentally compare some of these tools in various network scenarios. All of these approaches have the same purpose: the file has to be identical in the origin and in the target side, and they need a bidirectional link. On a congested network or on a noise link, the transfer time is unpredictable. In this work, we present a method with a predictable transfer time.

Unidirectional Links

Currently, satellite remains the dominant medium for contribution over long distances or remote locations, especially for urgent news feed contribution. DSNG usually has unidirectional video data link for transmitting video information from an origin point to the broadcast central station (1-to-1 transmission).

For file distribution over unidirectional links, there are various file transfer protocols based on retransmission patterns of the same file. The Reliable Multicast Transport (RMT) IETF Working Group deals with the standardization of reliable one-to-many multicast transport protocols. The Asynchronous Layered Coding (ALC) [15] does not require any type of feedback from the receivers, and the data are encoded using FEC codes. Repetitions of symbol transfer guarantee the integrity of the file at the expense of diminished effectiveness in the bandwidth. The Nack Oriented Reliable Multicast (NORM) [16] retransmits only the damaged parts from some of the receptors which send signals of Negative Acknowledgments (NACK) over damaged blocks and the File Delivery over Unidirectional Transport (FLUTE) [17], based on the ALC protocol, with the extension to be used in any type of transmission channel (unidirectional or not) presents metadata which complete the image signal itself (i.e. File name, codec, etc.).

These protocols have several drawbacks for transmission of larger files over a unidirectional and noisy channel:

- Bandwidth wastage because the cycled retransmission of the file.
- Unreliability of the transfer in a specific time, because one block can have errors in all his transmissions.
- Bad response for 1-to-1 large files transfer.

In this work, we present an approach for solving unidirectional transmission problems in with the data is distributed among two lines.

JPEG2000

In a first decomposition step, the 2-D forward Discrete Wavelet Transform (DWT) decomposes an image into LL – Lower resolution version of image, HL – Vertical edge data, LH – Horizontal edge data and HH – Diagonal edge data sub-bands (1 layer). The LL sub-band is a coarse approximation of the original image, containing its overall geometric structure (low-frequency), while the other three sub-bands contain the high-frequency components of the image. After the first-step wavelet transform, four sub-band are created (LL_0 , HL_0 , LH_0 , and HH_0), where LL_0 is the lowest frequency sub-band. Successive decompositions are applied on the low frequency sub-band of previous step (LL_x) sub-band of layer x in order to create LL_{x+1} , HL_{x+1} , LH_{x+1} , and HH_{x+1} of $x+1$ (see Fig. 2).

Sub-Bands and Image Quality

At the end of the transforms, the LL sub band is essentially a thumbnail of the original image which contains all the most significant information. The multiple resolutions of the transform give the ability to extract and decode a smaller resolution image from the code stream without decompression. It's simply a matter of only decoding or transmitting the sub-bands up to the desired resolution. The fact that the most important information is in the LL sub band implies that in a lossy transmission environment, errors in the other 80% of the code stream will have little visual impact. This reduces the need to sacrifice bandwidth on further error correction, as well as the extra development effort needed to implement it. In Fig. (3), one can see how the sub-bands add quality to the codified image from LL_2 to HH_0 .

Codification

In transform based lossy image compression systems (as JPEG 2000), the extent of compression is typically controlled by providing the compression system with a target, either a net compressed file size or an indication of how much loss of video quality is acceptable. For example, at the user interface level, an image quality number of 1-10. In order to maintain consistency across images, the subjective inputs can be mapped to objective estimates of image distortion, such as peak-signal-to-noise-ratio (PSNR). In this section, we present some quality indication can be input as qualitative description ('high quality', 'medium quality', or 'low quality').

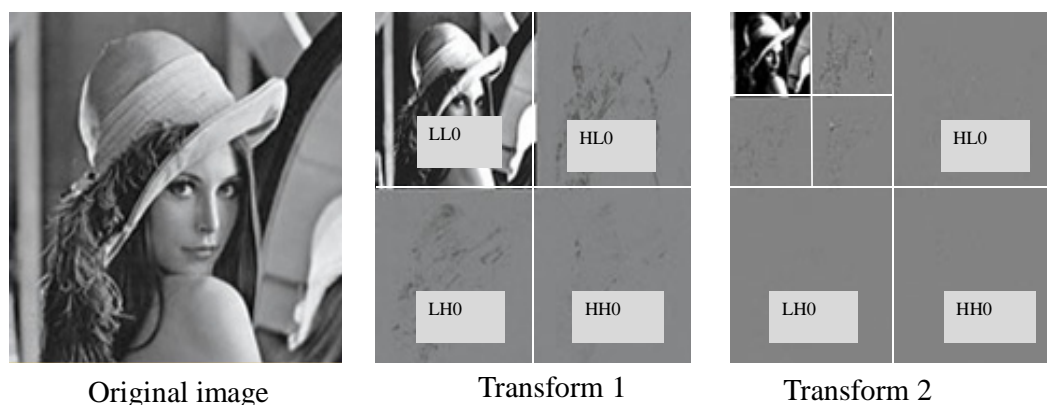


Fig. (2). JPEG2000 Image decomposition.

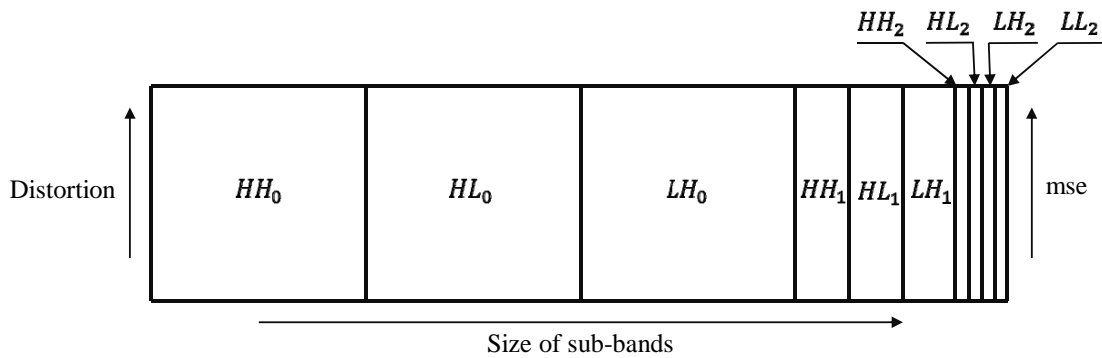


Fig. (3). RMSE evolution with the JPEG2000 sub-bands size for 3 decomposition levels.

Rastogui [18] presents systems and techniques relating to determining scalar quantizers for a signal, for example, determining scalar quantizers to generate an image of a desired distortion in wavelet-based compression.

Jones [19] presents a method for controlling the amount of compressed data when one or more viewing condition parameters are specified. A quantizer step size for each frequency subband is determined that minimizes an amount of data that must be encoded for a compressed representation of the image by using one or more viewing condition parameters and a model that characterizes the human visual system. The image is compressed using the quantizer step size for each frequency subband to produce an amount of compressed data. The presence of a pre-specified maximum amount of compression data is determined. The amount of compressed data is compared to the pre-specified maximum amount. Subsequent discarding of least visually relevant compressed data reduces the amount of compressed data.

Resman [20] introduces a system for managing storage space on an electronic storage medium in which a file format for stored data allows for progressive deletion of low-significance data, for example in a video or audio file, while allowing the remaining portions of the file to be subsequently retrieved. The file format allows for the ready deletion of low-significance data without having to open, edit and subsequently rewrite the data. Furthermore, rules-based algorithms for the deletion of low-significance data allow a user to store and progressively delete such low-significance data in accordance with time parameters, available storage space, without having to delete the full file.

A predictive method [21], and apparatus are directed towards computing minimum mean squared error (MMSE) predictive-transform (PT) source coding integrated with subband compression to further improve the performance of low bit rate MMSE PT source coders. A desirable byproduct of the advanced scheme is that the incorporation of joint optimum prediction and transformation from subband to subband is suited to its integration with JPEG2000 to yield even higher compression levels while producing an outstanding objective as well as subjective visual performance. Yasurito used the same idea in the client side [22].

Quast presents a patent based on a user defined ROI the method transcodes HD frames into images in moderate resolution with a Region Of Interest (ROI) in HD resolution [28].

Unequal Error Protection

Error in layer x produces less mean square errors (mse) in the image than errors in layer $x+1$, based on this characteristic the Unequal Error Protection (UEP) methods protect more some layers than others. There are a lot of works exploiting UEP for JPEG2000 [23-27] the problem I the data is not totally protect and all of these methods are applied to video stream (no file). This paper presents a method similar to UEP and predictive transform applied for file transfer though noise channels.

PROPOSED SYSTEM

The basic idea is to protect most important data from errors; the proposal has similarities with UEP but with vital data fully protected. Another difference is while UEP work with video_stream, this approach uses compressed files for transmission process.

Two Files

The video file-transfer method is based on splitting the file into two separate files, "key" and "support" files, reliable and unreliable transfer, respectively. The key file will contain at least: all the boxes header, Code-Stream box marks and the most important video data. The less important video data will be carried into the "support" file. A classic video stream consists in to send of all the data over an unreliable transfer with different redundant codes, whit this approach we get a fully reliable transfer of at least all the heathers and marks data.

In the Fig. (4), we see two lines, key and support file size evolution. Depending on the situation, users can choose their own ideal pair of files. On the vertical side, where the possible combinations of sizes between files are placed; as can be observed, the first option on the left is when in the key file has only the data corresponding to marks and headers, all the bitstream of the compressed image go into the secondary file. The last option corresponds with a classic ftp transfer in which the entire file is placed in one file (key file).

In a first filter, the different boxes will be sent which is not Contiguous Code-stream Box, under the key file while the content of the referred box will be:

- The headers will go into the key file.

- The content of all the marks except the Start Of Data mark (SOD) content will go into the key file.
- The content from the SOD mark will go into the key or support file depending on the balanced size division.

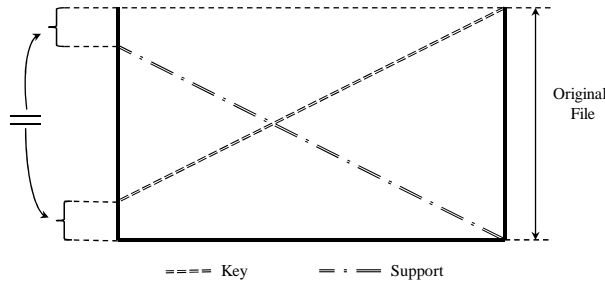


Fig. (4). Different values for Key and Support files.

In the case that the information of the different layers is coded with the same number of bits, with an image format of 16/9 and for two levels of decomposition, in a first decomposition:

$$S(LL_0) = S(HL_0) = S(LH_0) = S(HH_0) = \frac{1}{4} S(file)$$

where S indicates the file size. In a second decomposition:

$$S(LL_1) = S(HL_1) = S(LH_1) = S(HH_1) = \frac{1}{4} S(LL_0)$$

In this way, each one of the decompositions of level two will be 16 times less than the original image, this way the code size will be composed of:

$$S(image) = S(LL_1) + S(HL_1) + S(LH_1) + S(HH_1) + S(LL_0) + S(LH_0) + S(HH_0)$$

Being:

$$S(LL_1) = S(HL_1) = S(LH_1) = S(HH_1) = \frac{1}{16} S(file)$$

This way the stored of the different sub-bands can be separated by key or support files. In general, it can be stated that the size of the stream of an image with N levels has:

$$S(file) = S(headers) + S(LL_{N-1}) + \sum_{n=N-1}^{n=0} S(LH_n) + S(HL_n) + S(HH_n)$$

where $S(image)$ is the size of the original file. If the bandwidth required a cut-off in layer j , the file sizes for key and support files are:

$$S_k = S(headers) + S(LL_{N-1}) + \sum_{n=N-1}^{n=j-1} S(LH_n) + S(HL_n) + S(HH_n)$$

$$S_s = \sum_{n=j}^{n=0} S(LH_n) + S(HL_n) + S(HH_n)$$

For transmission task, the key file is to be sent to the destination by a reliable transmission (e.g. ftp, tsunami, etc.), while the support file is to be sent by unreliable transfer or streaming mode, UDP or unidirectional link (see Fig. 5). On the receiving side, once the main file is received, the quantity of bytes remaining to be received can be known by confirming the box start arrival without having received all the contiguous code-stream information, in this way the empty spaces will fill with the information which arrives in the data file. The support file in the central station could be different from the remote office, but the Key file has to be the same. The errors of the 'original' file in the central station could be treated with some type of concealment techniques.

If we want to minimize the transfer time $TT(min)$:

$$TT(min) = TT(Support File) = TT(Key File)$$

Depending of the bitrates of each channel (reliable or unreliable), we have the sizes of the files. If we want to maximize the quality in a period of time (PoT, we have to add more data in the key file:

$$PoT = TT(Key File) > TT(Support File)$$

If we want minimize the transfer time $TT(min)$:

$$TT(min) = TT(Support File) = TT(Key File)$$

Application

The program reads the JPEG2000 original file placing it into a buffer with updates when data is needed. The program scans the entrance stream recognizing the headers and the fields of the box size to be analyzed. A series of files is generated according to the sub-bands of each layer (see the algorithm in Fig. (6)).

- 1- A box is extracted and is observed if it is not code-stream.
- If it is not, it is stored in the key file; another box is extracted and returns to point 1.

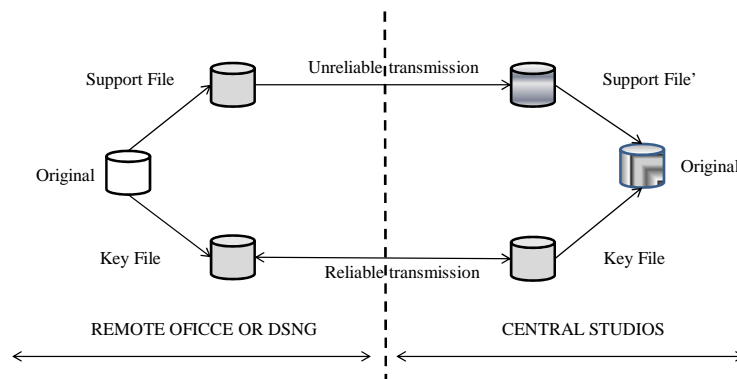


Fig. (5). Two files transfer model (TFTM).

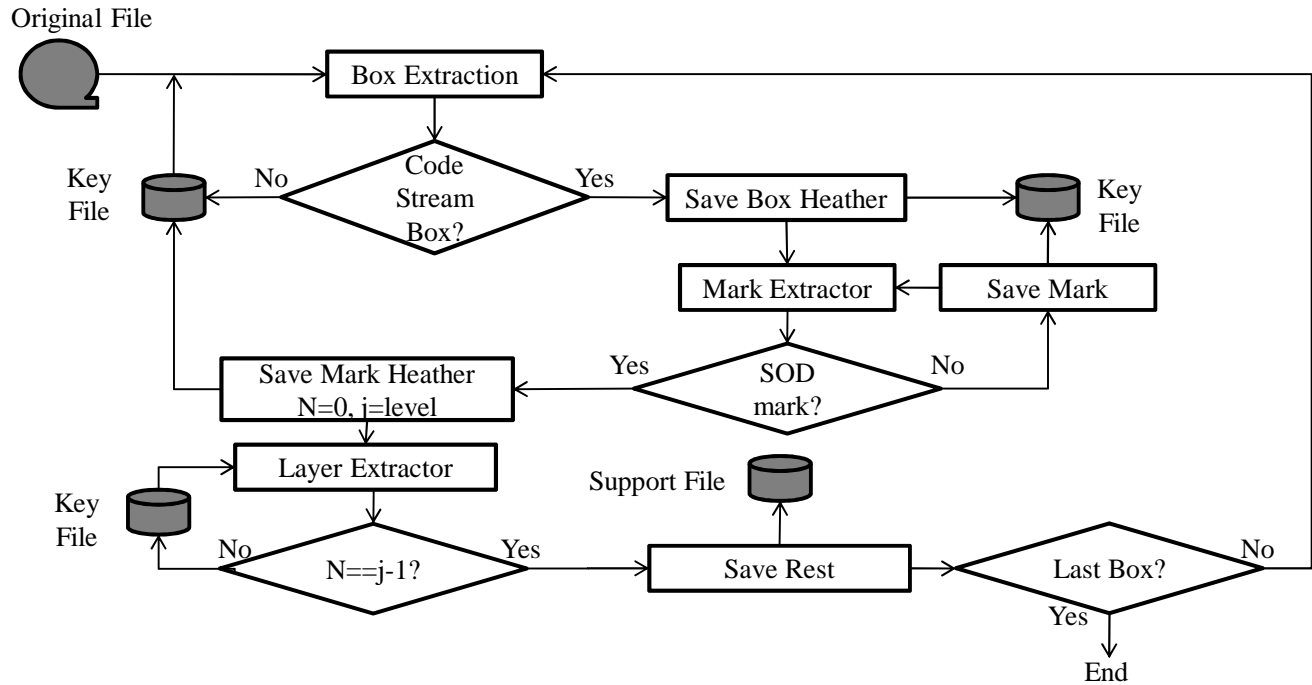


Fig. (6). Application Flowchart.

- If it is, the entire header is stored in key file.
- 2- A mark is extracted and is observed if it is SOD (Start of data).
- If it is not, all the mark is stored in the key, another mark is extracted and one returns to point 2.
- If it is, all the header of the mark is stored in key file and a counter is initialized.
- 3- One layer is extracted:
- If counter > division, the entire layer is stored in the key file, another layer is extracted, the counter is upgraded and one returns to point 3.
- If it is the equal, the rest of the box is stored in the support file.
- If it is the last box, the files are closed and the process is finished, if there are more boxes, going to step 1.

Upon reception, the process to conform the file is similar to decomposition. First, one reads the received as key file and it is copied into the final file until a code-stream box is found in which case it is to be copied to the final file until the beginning of another box is found, it is the moment in which it must be copied from the received as support file the quantity of bits corresponding to the difference between the size which appears in the header of the code-stream box and the size received on the key one. This operation is repeated until the treatment of the files is completed.

SCENARIO

The image sequence used is provided by European Broadcaster Union EBU on its ftp server. The image is

composed of a sequence of high definition fixed images to be tested with systems. They are provided in RGB format and the available coder in Open JPEG2000 (image_to_J2K.exe) was used together with tools for sequencing of images in a movement. The images are coded in four layers numbered from 0 to 3; the progression order is layer-resolution-component-position (LRCP) progressive.

Image

For each image, the size of heathers' boxes and marks is 153 Bytes (0.46 MB for a video of 1 minute). As we mentioned above, it is necessary to use a set of two files for each level, the general combinations are shown in Fig. (7), the columns represents the different set of test file pairs, for level 1 the key file has the heathers and marks while the support file contain all the image data.

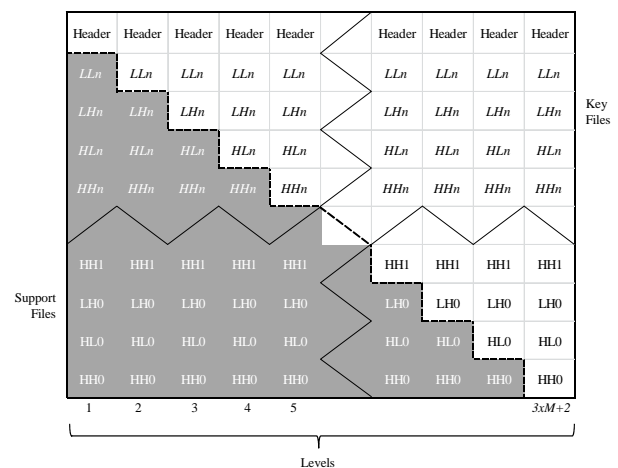


Fig. (7). Levels composition.

Table 2. Size of the Files (1 Minute) and rmse and psnr of a Frame in Case of Error in Each Level (50fps).

	Levels	Key(MB)	Support(MB)	rmse	psnr
Init (LL3)	1	0.46	187.04	56.20	31.72
	2	1.19	186.31	28.08	40.41
Layer 3	3	1.92	185.58	27.81	40.64
	4	2.65	184.85	27.79	40.66
	5	3.38	184.12	27.63	40.66
	6	6.31	181.19	8.21	47.35
Layer 2	7	9.23	178.27	6.61	62.38
	8	12.15	175.35	2.43	62.56
	9	23.84	163.66	2.36	65.62
Layer 1	10	35.53	151.97	2.36	65.62
	11	47.22	140.28	2.36	71.00
	12	93.98	93.52	1.09	77.52
Layer 0	13	140.74	46.76	1.19	79.68
	14	187.50	0.00	0	∞

This way, for each image different levels of separation in each of the files (key and support) are obtained; this separation is shown in a Table 2 where the key and support files are shown. On an initial level, the key file is composed only with heathers. On level two, the bitstream corresponding to LL_3 have been added; as levels increase, layers are added up to level 14 where all image bitstream is in the key file. For the bitstream corresponding to the support file, reading errors have been added to each of the subbands, with the peak signal noise ratio (psnr) generated from both files as shown in column "psnr".

The evolution of the psnr can be observed in Fig. (8). with respect to the sizes of the files.

Networks

Tests of file transfer have been performed from two different networks to be able to see the real rate of transfer between two terminals.

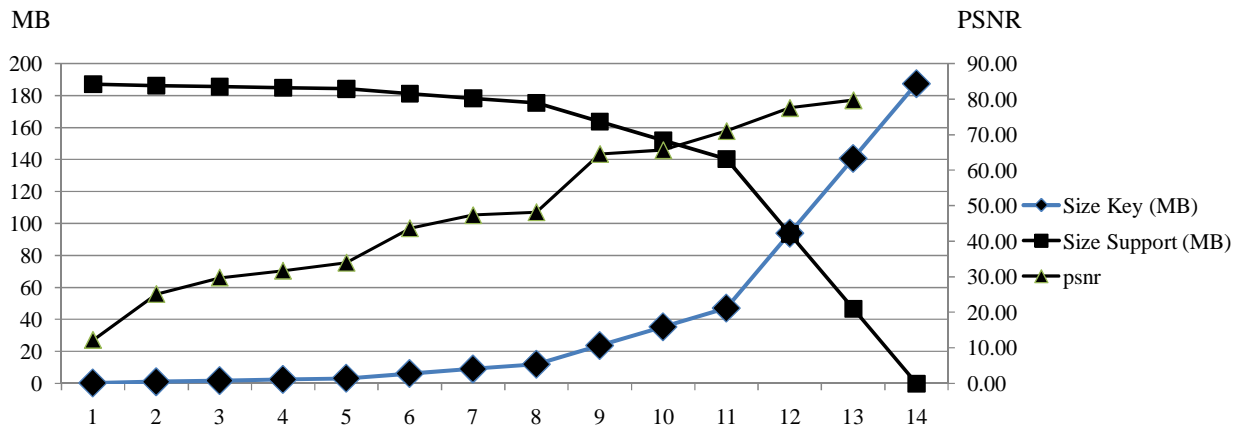
For bidirectional test, the network corresponding to Fig. (9). was used in which there is an original station in Madrid with a video codifier and a terminal where a series of videos are stored to be sent. The network which links the Madrid station with the destination is switched-packet network with QoS which connects Madrid to Sevilla at a speed of 40 Mbps. In the receptor stations, we had a terminal to receive the files and one to decode in order to see the video which arrives by streaming.

The accomplished transfer of files are those shown in table where the volume of sent files and the time taken for transfer are shown, with these data real transfer data can be drawn, as far as effective speed as well as percentage of bandwidth (see Table 3).

For unidirectional link, we use a satellite link with several bitrates of uplink, and one UMTS line (see Fig. 10).

METHOD RESPONSE

We used a classical one minute news file codified at 25

**Fig. (8).** PSNR evolution with key and support file sizes.

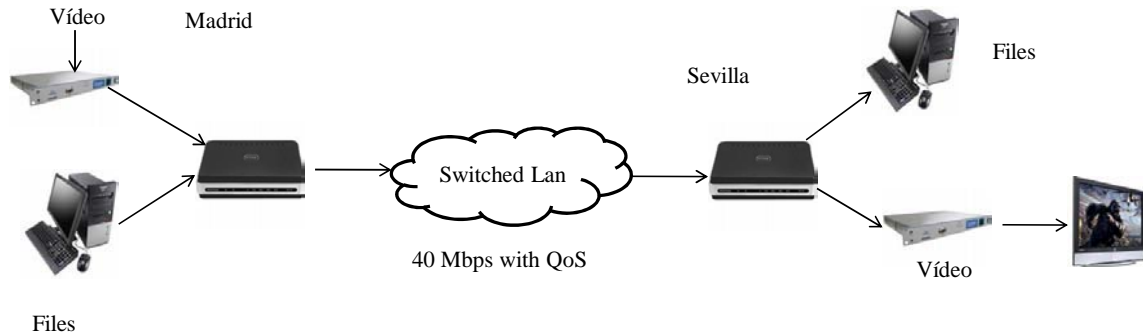


Fig. (9). Network-switched used in the tests, consisting of an Wan link of 40 Mbps between Madrid y Sevilla.

Mbps, the file size is 187,5 Mbytes.

Table 3. Network Response.

Size (Bytes)	T (SEG)	BPS	%
513000	130.00	3156923	7.89%
986000	2.50	3155200	7.89%
1363000	3.00	3634667	9.09%
2770000	6.00	3693333	9.23%
4176000	9.50	3516632	8.79%
5729871	13.28	3451293	8.63%
8092784	19.15	3381396	8.45%
10783982	24.56	3512397	8.78%

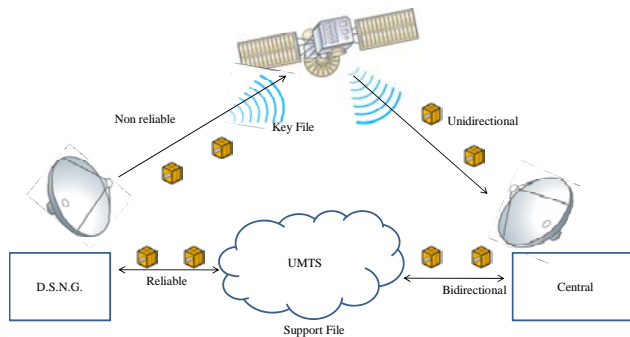


Fig. (10). Two line transmission: unidirectional (high bitrate) and bidirectional (low bitrate).

Bidirectional Link

We can see the comparative results between this model with streaming and copying files (see Table 4). For the 40Mbps switched-network mentioned above, the ftp transfer needs 187 seconds while in live or streaming mode the transfer will occur as live event (1 minute). The transfer time (TT) is 38 seconds (level1) because it can use the entire bandwidth in order to transmit the data, with a transmission of the level 1 form, we have fully protected all the headers. With a transmission of 63s, we are protecting headers and the last layer (the most important for the final image quality). With a very few increases of transmission time (VS streaming), we get safe image quality compare with streaming and transmission time compare with ftp.

In Fig. (11). It can be observed how the PSNR evolutions with respect to the transfer time according to the different levels and the total transfer time. At the beginning of the curve, it can be observed that with a small increase in transfer time, a very pronounced lowering of mse is achieved; at the end of the curve, the transfer time increases are not reflected in a great lowering of average quadratic error.

Unidirectional Link

In this case, the support file will be transmitted by the satellite link, while the key file is transferred thought a UMTS line. The satellite link has a 15Mbps unidirectional link (as used for news live events). In this type of transmission, broadcasters have to adapt the image quality to the line speed, with the processes shown at the very beginning of this article with the loss of quality resulting from these processes.

Again, we can observe the comparative results between this model and streaming (see Table 5). With a level 2 transmission, we are protecting the satellite link, all the headers and the LL₃ sub-band (99 sec. transmission time), this level has the minimum TT with this lines characteristics. We can maximize the image quality, if we have a five minutes time window for the transfer, we can use the level 5 assuring the entire layer 3.

In Fig. (12). it can be show the response graphic, and it is similar with bidirectional links with other values.

CURRENT AND FUTURE DEVELOPMENTS

The file transfer method proposed in this article presents an alternative way to transport stored programs between remote offices to central studios instead of traditional methods inherited from analogical transmissions. Old workflows increase the operational cost and degrade the image quality. With this solution Broadcasters receives benefits:

- It makes a file transfer instead a streaming, by this way it is possible to save human processes with the associated cost.
- Optimizes image quality with time available for transmission because this system can choose a set of files in order to protect the maximum data in a specific period of time.

Table 4. Transmission Time of jpeg2000 file with 1 Minute of Video with a Bidirectional Link and at 40 mbps Bandwidth.

	Lev.	Key	TTkey	Support	TTSup	TT	Rmse	psnr
Initial	1	0.46	1.08	187.04	37.4075	38.48	56.20	31.72
	2	1.19	2.78	186.31	37.2614	40.04	28.08	40.41
Ly.3	3	1.92	4.48	185.58	37.1153	41.59	27.81	40.64
	4	2.65	6.18	184.85	36.9691	43.15	27.79	40.66
	5	3.38	7.88	184.12	36.8230	44.70	27.63	40.66
	6	6.31	14.68	181.19	36.2385	50.92	8.21	47.35
Ly. 2	7	9.23	21.48	178.27	35.6540	57.13	6.61	62.38
	8	12.15	28.28	175.35	35.0695	63.35	2.43	62.56
	9	23.84	55.48	163.66	32.7316	88.22	2.36	65.62
Ly. 1	10	35.53	82.69	151.97	30.3936	113.08	2.36	65.62
	11	47.22	109.89	140.28	28.0556	137.95	2.36	71.00
	12	93.98	218.71	93.52	18.7038	237.41	1.09	77.52
Ly. 0	13	140.74	327.52	46.76	9.3519	336.87	1.19	79.68
	14	187.50	436.33	0.00	0.0000	436.33	0	∞

- As a result of file transfers, it is not necessary to adapt the image quality to the line speed, this adaptation degrades the image quality, as results this method conserve the original bitrate of the image.

Broadcasters can use the same communications networks and equipment that actually they have in order to adapt this method in their video transfer system as we show with two video link type without additional infrastructure cost.

This approach try to adapt the video production requirements based on the video file data structure instead of the global needs. Actually, we are trying to use the same idea with video servers. With this architecture, an economical method for the storage of multimedia professional content applied to JPEG2000 files can be formulated. The adaptation of the method can use several types of RAID configuration and the video file data could be decomposed into different files in order to correlate the redundancy with importance of

the data. The global storage can be used in all the different tasks of the production workflow saving disk space, redundant files and transfers procedures.

As future developments the idea can be exported to other video format such as MPEG2, DVCPRO, etc. with the support of all the professional video compression formats, the application can be used for real production processes.

ACKNOWLEDGEMENTS

Thanks to my colleagues of Antena 3 Televisión and Radio Televisión Española(RTVE). We've all had good, bad and very bad times during the transformation of old workflows systems based in video tapes to the new production based in video files. Thanks to the reviewers of this journal, they have been very patient and worked hard to help me in the preparation of this article.

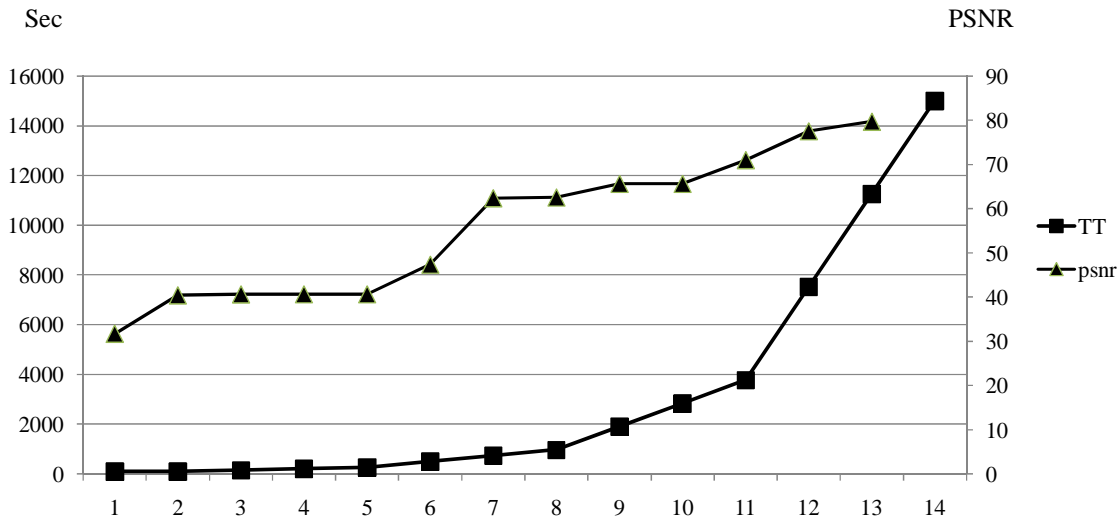
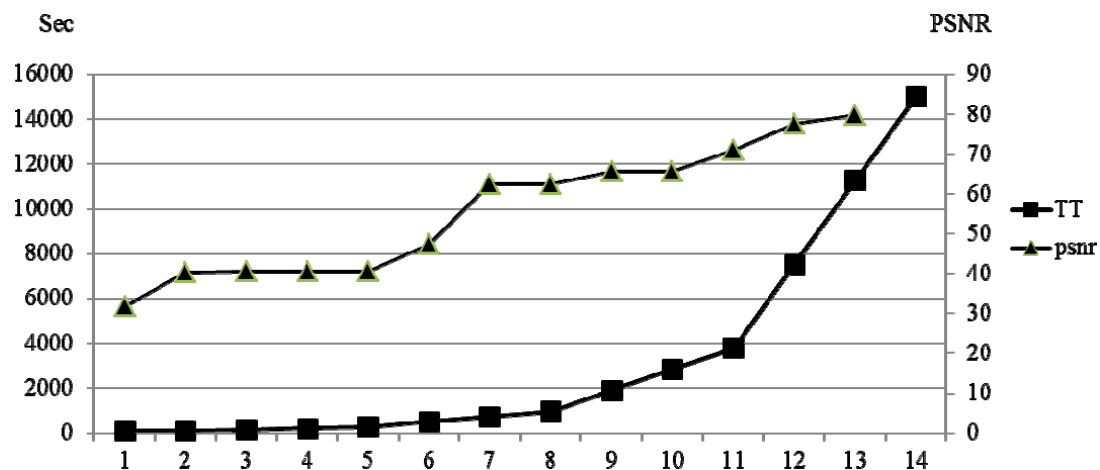
**Fig. (11).** Mse, psnr evolutions respect to the transfer time (TT) according to the different levels.

Table 5. Transmission Time of jpeg2000 File with 1 Minute of Video with a Unidirectional Link (15 Mbps) Over a UMTS Link.

	Lev	Key	TTkey	Support	TTSup	TT	psnr
Initial	1	0.46	37	187.04	99.7533	99	31.72
	2	1.19	95	186.31	99.3637	99	40.41
Ly. 3	3	1.92	153	185.58	98.9740	153	40.64
	4	2.65	212	184.85	98.5843	212	40.66
	5	3.38	270	184.12	98.1947	270	40.66
	6	6.31	504	181.19	96.6360	504	47.35
Ly. 2	7	9.23	738	178.27	95.0774	738	62.38
	8	12	972	175.35	93.5188	972	62.56
	9	23	1907	163.66	87.2842	1907	65.62
Ly. 1	10	35	2842	151.97	81.0496	2842	65.62
	11	47	3777	140.28	74.8150	3777	71.00
	12	93	7518	93.52	49.8767	7518	77.52
Ly. 0	13	140	11259	46.76	24.9383	11259	79.68
	14	187	15000	0.00	0.0000	15000	∞

**Fig. (12):** Mse evolution with satellite and UMTS link.

CONFLICT OF INTEREST

None

REFERENCES

- [1] ISO/IEC, 15444-1:2000 Information technology - JPEG2000 - image coding system - Part 1: Core coding system, International Standard ISO/IEC 15444-1 Std., 2000.
- [2] ISO/IEC 13818-1:2000 Information technology -- Generic coding of moving pictures and associated audio information: Systems.
- [3] ETSI. *Digital Video Broadcasting (DVB); Frame structure channel coding and modulation for a second generation digital terrestrial television broadcasting system (DVB-T2)*. Ed: European Broadcasting Union 2009.
- [4] ETSI. *Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications (DVB-S2)*. Ed: European Broadcasting Union 2009
- [5] ETSI. *Broadcasting to Handhelds; Digital Terrestrial Mobile TV*. Ed: European Broadcasting Union 2011
- [6] ETSI. *Digital Video Broadcasting (DVB); Frame structure channel coding and modulation for a second generation digital transmission system for cable systems (DVB-C2)*. Ed: European Broadcasting Union 2011
- [7] J. Postel and J. Reynolds, *File transfer protocol (FTP)*, Available at: <http://tools.ietf.org/html/rfc959>, 1985.
- [8] J.M. Barbero and E. Santos, "Method for multimedia file transmission", PCT/ES000382, 2009.
- [9] J.M. Barbero and E. Santos, "Dual method for multimedia file transmission", PCT/ES 000203, 2010.
- [10] F.G.B. De Natale, "Error concealment in digital video", In: *Digital Image Sequence Processing, Compression, and Analysis*, CRC press 2004.
- [11] J. Postel and J. Reynolds, "File transfer protocol (FTP)", Available at: <http://tools.ietf.org/html/rfc959>, 1985.
- [12] M. Noro, F. Tameshige, K.-I. Baba and S. Shimojo, "Performance Evaluation of Advanced High-Speed Data Transfer Methods in Long-Distance Broadband Networks", *22nd Int. Conf. on Advanced Information Networking and Applications - Workshops, AINAW 2008*, Okinawa, Japan, 25-28 March, 2008, pp. 533-40.

- [13] J. Zhang and R. D. McLeod, "A UDP-Based File Transfer Protocol (UFTP) with flow control using a rough set approach," *IEEE Trans. Networking*, 2005.
- [14] C. Neumann, V. Roca and R. Walsh, "Large scale content distribution protocols", *SIGCOMM Comput. Commun.*, Philadelphia, PA, USA, Comput. Commun. Rev., vol. 35(5), pp. 85-92, 2005.
- [15] M. Luby, M. Watson, J. Gemmel, L. Vicisano, L. Rizzo and J. Crowcroft, "Asynchronous Layered Coding (ALC) Protocol Instantiation", IETF, February 2007.
- [16] B. Adamson, C. Bormann, M. Handley and J. Macker, "Negative-acknowledgment (NACK)-Oriented Reliable Multicast (NORM) Protocol", IETF. RFC 3940, 2004.
- [17] T. Paila, R. Walsh, M. Luby, R. Lehtonen and V. Roca, "FLUTE - File Delivery over unidirectional transport", IETF, January 2007.
- [18] A. Rastogi, "Determining scalar quantizers for a signal based on a target distortion", U.S. Patent 7639886, 2009.
- [19] P.W. Jones, "Method for controlling the amount of compressed data", U.S. Patent 7729551, 2010.
- [20] P. W. Jones, "Method and apparatus for progressively deleting media objects from storage", U.S. Patent 7751628, 2010.
- [21] H. Feria, "Predictive-transform source coding with subbands", U.S. Patent 2007079469, 2007.
- [22] Yasuhiro, "Image processing system that internally transmits lowest-resolution image suitable for image processing", U.S. Patent 7349585, 2008.
- [23] K. Quast, A. Kaup and H. Hampel, "Method and system for transcoding regions of interest in video surveillance", U.S. Patent Appl. 20110051808A1, 2011.
- [24] M.W. Marcellin and A. Bilgin, "JPEG2000: Highly scalable image compression," *Int. Conf. on Information Technology: Coding and Computing, (ITCC '01)*, Baltimore, USA, 2001, pp. 268-72.
- [25] M. Kurosaki, K. Munadi and H. Kiya, "Error concealment using layer structure for JPEG2000 images", *Asia-Pacific Conf. on Circuits and Systems, Phuket, Thailand*, 2002, vol.1, pp. 529-34.
- [26] Y. Qinghua and C. Miao, "Study of unequal error protection method based on JPEG2000 for surveying & mapping image", *Int. Conf. on Computer Science and Software Engineering*, Wuhan, China, 12-14 Dec, 2008, vol.2, pp. 951-4.
- [27] J. Serra-Sagrista and F. Auli-Llinas, "JPEG2000: Quality scalability and windows of interest transmission", *Signals, Systems and Computers, ACSSC*, California, USA, 4-7 Nov, 2007, pp. 920-4.
- [28] G. Zhaohui, Y. Nishikawa, R.Y. Omaki, T. Onoye and I. Shirakawa, "A low-complexity FEC assignment scheme for motion JPEG2000 over wireless network," *Int. Conf. on Consumer Electronics, ICCE '06*, Digest of Technical Papers., San Diego, USA, 7-11 Jan, 2006, pp. 391-2.